

[illegible]

3

Sy

MT  
ME

MT

MT

MT

MT

MT

MT  
MT

MT

MT  
MT

MT  
MT

MT  
MT

MT  
MT

MY  
MT

MY  
MT

MT  
MTMT  
MTMT  
MTMT  
MT

MT

MT  
MT

MT

MT

MT

M1

W1  
M1  
22

M1

M1

1

1

1

1

1

[illegible]

(2) 57  
(3) 87  
(4) 146

DECLARATIONS  
MT\$HFLOOR - greatest integer Hfloating routine  
MT\$HFLOOR\_R7 - greatest integer Hfloating routine



```
0000 1 .TITLE MTH$HFLOOR - Greatest integer routine for H floating
0000 2 .IDENT /1-002/ ; File: MTHHFLOOR.MAR EDIT: RH1002
0000 3
0000 4
0000 5 *****
0000 6 *
0000 7 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 8 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 9 * ALL RIGHTS RESERVED.
0000 10 *
0000 11 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 12 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 13 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 14 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 15 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 16 * TRANSFERRED.
0000 17 *
0000 18 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 19 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 20 * CORPORATION.
0000 21 *
0000 22 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 23 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 24 *
0000 25 *
0000 26 *****
0000 27
0000 28
0000 29 ++
0000 30 FACILITY: Math Library
0000 31
0000 32 ABSTRACT:
0000 33
0000 34 This routine finds the largest integer less than the input
0000 35 value, i.e. it truncates toward negative infinity
0000 36 for data type H_floatin.
0000 37
0000 38 ENVIRONMENT: User Mode, AST Reentrant
0000 39
0000 40 --
0000 41 Author: John Sauter, Creation date: 27-JUL-1979
0000 42
0000 43 MODIFIED BY:
0000 44
0000 45 VERSION 00
0000 46 1-001 - Original, from MTH$GFLOOR.
0000 47 1-002 - CALL entry was modified to return the result in the address
0000 48 specified by the leftmost argument in order to conform to the
0000 49 calling standard for return values larger than 64 bits. The
0000 50 original version returned the results in R0-R3.
0000 51 JSB entry was modify to correct typos: The last two operands
0000 52 of the EMODH instruction and the operand of the TSIH instruction
0000 53 were changed from R2 to R4.
0000 54 Comments were changed to eliminate an erroneous calling sequence.
0000 55 RNH 9-DEC-1980
```

```

0000 57      .SBTTL  DECLARATIONS
0000 58  ::
0000 59  :: INCLUDE FILES:
0000 60  ::
0000 61  ::
0000 62  ::
0000 63  :: EXTERNAL DECLARATIONS:
0000 64  ::
0000 65      .DSABL  GBL                      ; Prevent undeclared
0000 66                                          ; symbols from being
0000 67                                          ; automatically global.
0000 68  ::
0000 69  :: MACROS:
0000 70  ::
0000 71  ::
0000 72  ::
0000 73  :: EQUATED SYMBOLS:
0000 74  ::
0000 75  ::
0000 76  ::
0000 77  :: OWN STORAGE:
0000 78  ::
0000 79  ::
0000 80  ::
0000 81  :: PSECT DECLARATIONS:
0000 82  ::
0000 83      .PSECT _MTH$CODE PIC, USR, CON, REL, LCL, SHR, -
0000 84      EXE, RD, NOWRT, LONG
0000 85

```

```
0000 87 .SBTTL MTH$HFLOOR - greatest integer H_floating routine
0000 88 :++
0000 89 : FUNCTIONAL DESCRIPTION:
0000 90 :
0000 91 : This routine finds the floor by truncating, and then if the
0000 92 : input value is negative and not an integer subtracting 1.
0000 93 :
0000 94 : CALLING SEQUENCE:
0000 95 :
0000 96 : CALL MTH$HFLOOR (result_int.wh.r, input.rh.r)
0000 97 :
0000 98 : INPUT PARAMETERS:
0000 99 :
00000008 0000 100 : input_addr = 8 ; address of the H_floating number
0000 101 : ; to get the floor of
0000 102 :
0000 103 : IMPLICIT INPUTS:
0000 104 :
0000 105 : NONE
0000 106 :
0000 107 : OUTPUT PARAMETERS:
0000 108 :
00000004 0000 109 : output_addr = 4
0000 110 :
0000 111 : IMPLICIT OUTPUTS:
0000 112 :
0000 113 : NONE
0000 114 :
0000 115 : FUNCTION VALUE:
0000 116 : COMPLETION CODES:
0000 117 :
0000 118 : NONE
0000 119 :
0000 120 : SIDE EFFECTS:
0000 121 :
0000 122 : NONE
0000 123 :
0000 124 : --
0000 125 :
00FC 0000 126 .ENTRY MTH$HFLOOR, *M<R2, R3, R4, R5, R6, R7> ; entry point
0002 127
54 54 08 50 08 BC 70FD 0002 128 MOVH @input_addr(AP), R0 ; R0/R3 = input argument
0007 129 EMODH R0, #0, #1, R4, R4 ; R4/R7 = fraction_part (arg)
000E 130 SUBH2 R4, R0 ; R0/R3 = integer_part (arg)
0012 131
0012 132 BGTR 40$ ; if > 0, have correct answer
0014 133
0014 134 TSTH R4 ; look at fraction part
0017 135 BGEQ 40$ ; if > 0 then 0 < input < 1 and
0019 136 ; we have the correct answer
0019 137 ; if = 0 then input was integer
0019 138 ; and we have correct answer
0019 139
0019 140 SUBH2 #1, R0 ; subtract 1 from truncated
001D 141 ; negative non-integer
001D 142
04 BC 50 70FD 001D 143 40$: MOVH R0, @output_addr(AP) ; move result to output address
```



MTH\$HFLOOR  
1-002

- Greatest integer routine for H<sub>1</sub> floatin 16-SEP-1984 01:36:01 VAX/VMS Macro V04-00  
MTH\$HFLOOR - greatest integer H<sub>1</sub> floatin 6-SEP-1984 11:24:55 [MTHRTL.SRC]MTHHFLOOR.MAR;1 Page 4  
04 0022 144 RET (3)

MTH  
2-0

```
0023 146 .SBTTL MTH$HFLOOR_R7 - greatest integer H_floating routine
0023 147 :++
0023 148 : FUNCTIONAL DESCRIPTION:
0023 149 :
0023 150 : This is the JSB entry point to MTH$HFLOOR.
0023 151 :
0023 152 : CALLING SEQUENCE:
0023 153 :
0023 154 : JSB MTH$HFLOOR_R7
0023 155 :
0023 156 : INPUT PARAMETERS:
0023 157 :
0023 158 : R0 through R3 contain the input value
0023 159 :
0023 160 : IMPLICIT INPUTS:
0023 161 :
0023 162 : NONE
0023 163 :
0023 164 : OUTPUT PARAMETERS:
0023 165 :
0023 166 : R0 through R3 contain the result value
0023 167 :
0023 168 : IMPLICIT OUTPUTS:
0023 169 :
0023 170 : NONE
0023 171 :
0023 172 : FUNCTION VALUE:
0023 173 : COMPLETION CODES:
0023 174 :
0023 175 : NONE
0023 176 :
0023 177 : SIDE EFFECTS:
0023 178 :
0023 179 : NONE
0023 180 :
0023 181 :--
0023 182 :
0023 183 MTH$HFLOOR_R7::
0023 184 : entry point
54 54 08 00 50 74FD 0023 185 EMODH R0, #0, #1, R4, R4 ; R4/R7 = fraction_part (arg)
50 54 62FD 002A 186 SUBH2 R4, R0 ; R0/R3 = integer_part (arg)
002E 187
002E 188 BGTR 40$ ; if > 0, have correct answer
0030 189
0030 190 TSTH R4 ; look at fraction part
04 18 0033 191 BGEQ 40$ ; if > 0 then 0 < input < 1 and
0035 192 ; we have the correct answer
0035 193 ; if = 0 then input was integer
0035 194 ; and we have correct answer
0035 195
50 08 62FD 0035 196 SUBH2 #1,R0 ; subtract 1 from truncated
0039 197 ; negative non-integer
0039 198
05 0039 199 40$: RSB
003A 200
003A 201 .END
```



INPUT\_ADDR = 00000008  
MTH\$HFLOOR 00000000 RG 01  
MTH\$HFLOOR R7 00000023 RG 01  
OUTPUT\_ADDR = 00000004

! Psect synopsis !

PSECT name	Allocation	PSECT No.	Attributes
ABS	00000000 ( 0.)	00 ( 0.)	NOPICT USR CON
MTH\$CODE	0000003A ( 58.)	01 ( 1.)	PIC USR CON ABS REL LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	31	00:00:00.13	00:00:01.98
Command processing	128	00:00:00.51	00:00:07.16
Pass 1	72	00:00:00.54	00:00:01.82
Symbol table sort	0	00:00:00.00	00:00:00.00
Pass 2	50	00:00:00.47	00:00:02.05
Symbol table output	2	00:00:00.00	00:00:00.01
Psect synopsis output	2	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	287	00:00:01.69	00:00:13.07

The working set limit was 900 pages.  
2169 bytes (5 pages) of virtual memory were used to buffer the intermediate code.  
There were 10 pages of symbol table space allocated to hold 4 non-local and 2 local symbols.  
201 source lines were read in Pass 1, producing 11 object records in Pass 2.  
0 pages of virtual memory were used to define 0 macros.

! Macro library statistics !

Macro library name	Macros defined
_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	0

0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/DISABLE=(GLOBAL,TRACEBACK)/LIS=LIS\$:MTHHFLOOR/OBJ=OBJ\$:MTHHFLOOR MSRC\$:MTHHFLOOR/UPDATE=(ENH\$:MTHHFLOOR)



0262 AH-BT13A-SE  
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY